AMENDMENTS TO THE CLAIMS

This listing of claims will replace all prior versions, and listings, of claims in the application:

1. (Original) In an application service provision apparatus having an application service provision runtime library with multiple versions, a method of operation comprising:

loading the latest version of the runtime library at initialization of the application service provision apparatus, the latest version of the runtime library includes data associating an application with a required version of the runtime library;

during operation, receiving by a dispatcher a request for service for the application;

in response, determining by the dispatcher whether the required version of the runtime library used by the application is known to the dispatcher; and

if the version of the runtime library required by the application is not known to the dispatcher, inquiring by the dispatcher of the latest version of the runtime library to learn of the required version of the runtime library.

2. (Original) The method of claim 1, wherein said method further comprises the latest version of the runtime library informing the dispatcher which version of the runtime library is the required version of the runtime library, and the dispatcher caching the required version information.

3. (Original) The method of claim 2, wherein said method further comprises the dispatcher routing the request of service to the application to handle if the dispatcher is informed by the latest version of the runtime library that the required version of the runtime library is the latest version itself.

4. (Previously Presented) The method of claim 2, wherein said method further comprises the dispatcher determining whether the required version of the runtime library is

PAGE 4/13 * RCVD AT 3/22/2006 3:21:11 PM [Eastern Standard Time] * SVR:USPTO-EFXRF-5/22 * DNIS:2738300 * CSID:2062920460 * DURATION (mm-ss):04-02

loaded if the required version is an earlier version of the runtime library and, if the required earlier version of the runtime library is not loaded, loading the required earlier version.

5. (Original) The method of claim 4, wherein said method further comprises the dispatcher routing the request of service to the application to handle if the required earlier version of the runtime library is already loaded or upon loading the required earlier version of the runtime library.

6. (Original) The method of claim 1, wherein said method further comprises the dispatcher routing the request for service to the application to handle if the required version of the runtime library is known to the dispatcher.

7. (Original) In an application service provider apparatus having an application service provision runtime library with multiple versions, a method of operation comprising:

inquiring with the latest version of the runtime library to identify the version of the runtime library required by an application whenever the required version is not known; and

in response, the latest version of the runtime library responding with the required version.

8. (Original) The method of claim 7, wherein the method further comprises loading the latest version of the runtime library at initialization of the application service provider apparatus.

9. (Previously Presented) An apparatus comprising:

storage medium having stored therein programming instructions designed to implement a dispatcher on the apparatus to:

load a pre-determined version of a runtime library with multiple versions at initialization of the apparatus, the pre-determined version being the latest version,

receive a request for service for an application during operation,

PAGE 5/13 * RCVD AT 3/22/2006 3:21:11 PM [Eastern Standard Time] * SVR:USPTO-EFXRF-5/22 * DNIS:2738300 * CSID:2062920460 * DURATION (mm-ss):04-02

determine, in response, whether the version of the runtime library required by the application is known to the dispatcher, and

inquire with the latest version of the runtime library to learn of the required version of the runtime library if the version of the runtime library required by the application is not known to the dispatcher; and

at least one processor coupled to the storage medium to execute the programming instructions.

10. (Original) The apparatus of claim 9, wherein the dispatcher implemented by the programming instructions is further designed to receive from the latest version of the runtime library information on the required version of the runtime library, and in turn, cache the required version information.

11. (Original) The apparatus of claim 10, wherein the dispatcher implemented by the programming instructions is further designed to route the request of service to the application to handle if the dispatcher is informed by the latest version of the runtime library that the required version of the runtime library is the latest version itself.

12. (Original) The apparatus of claim 10, wherein the dispatcher implemented by the programming instructions is further designed to determine whether the required version of the runtime library is loaded if the required version is an earlier version of the runtime library, and if the required earlier version of the runtime library is not loaded, load the required earlier version.

13. (Original) The apparatus of claim 12, wherein the dispatcher implemented by the programming instructions is further designed to route the request of service to the application to handle if the required earlier version of the runtime library is already loaded or upon loading the required earlier version of the runtime library.

PAGE 6/13 * RCVD AT 3/22/2006 3:21:11 PM [Eastern Standard Time] * SVR:USPTO-EFXRF-5/22 * DNIS:2738300 * CSID:2062920460 * DURATION (mm-ss):04-02

14. (Original) The apparatus of claim 9, wherein the dispatcher implemented by the programming instructions is further designed to route the request for service to the application to handle if the required version of the runtime library is known to the dispatcher.

15. (Original) The apparatus of claim 9, wherein the storage medium further having stored therein programming instructions to implement the plurality of versions of the runtime library.

16. (Original) An apparatus comprising:

storage medium having stored therein programming instructions designed to implement a version of an application service provision runtime library, including the ability to receive an inquiry to identify the version of the runtime library required by an application, and in response, responding with the required version; and

at least one processor coupled to the storage medium to execute the programming instructions.

17. (Currently Amended) A method comprising:

accepting by a first resource consumer of a computer system, first allocations of a first plurality of portions of a shared resource, and tracking first points in time the first allocations were last used;

accepting by a second resource consumer of a computer system, second allocations of a second plurality of portions of the shared resource, and tracking second points in time the second allocations were last used;

conditionally requesting by a resource monitor of a computer system, the first and second resource consumers to provide said tracked first and second points in time, and the first and second resource consumers responsively providing the tracked first and second points in time as requested;

determining by the resource monitor which, if any, of the first and second allocations of the portions of the shared resource are to be released by the first and second resource consumers, and instructing the first and second resource consumers accordingly; and

PAGE 7/13 * RCVD AT 3/22/2006 3:21:11 PM [Eastern Standard Time] * SVR:USPTO-EFXRF-5/22 * DNIS:2738300 * CSID:2062920460 * DURATION (mm-ss):04-02

releasing by the first and second resource consumers, selected ones of the first and second allocations as instructed.

18. (Original) The method of claim 17, wherein the resource monitor conditionally requests the first and second resource consumers to provide said tracked first and second points in time when aggregated allocations of the shared resource reach a pre-determined threshold.

19. (Previously Presented) The method of claim 17, wherein said determining by the resource monitor comprises ordering said provided first and second points in time, and selecting a number of the least recently used allocations to be released to bring an aggregate of the allocations to at most a predetermined threshold.

20. (Previously Presented) In an apparatus having a shared resource consumer, a method of operation of the shared resource consumer, comprising:

accepting allocations of a plurality of portions of a shared resource;
tracking points in time the allocations were last used;
receiving a request to provide the tracked points in time;
in response, providing the tracked points in time as requested;
receiving instructions to release selected ones of the allocations; and
releasing the specified allocations as instructed.

21. (Original) The method of claim 20, wherein the apparatus is an application service provision apparatus, and the shared resource consumer is an application requiring application service provision runtime library support.

22. (Original) The method of claim 20, wherein the apparatus is an application service provision apparatus, and the shared resource consumer is a function of an application service provision runtime library.

PAGE 8/13 * RCVD AT 3/22/2006 3:21:11 PM [Eastern Standard Time] * SVR:USPTO-EFXRF-5/22 * DNIS:2738300 * CSID:2062920460 * DURATION (mm-ss):04-02

23. (Previously Presented) In an apparatus comprising a shared resource monitor, a method of operation of the shared resource monitor, comprising:

conditionally requesting a plurality of shared resource consumers to provide corresponding tracked plurality points in time, where corresponding plurality of portions of a shared resource allocated to the plurality of shared resource consumers were last used; and

determining which, if any, of the plurality of allocations of the portions of the shared resource are to be released by the plurality of shared resource consumers, and instructing the plurality of shared resource consumers to release selected ones of the plurality of allocations accordingly.

24. (Previously Presented) The method of claim 23, wherein said conditional request is made when aggregate allocations of the shared resource reach a pre-determined threshold.

25. (Original) The method of claim 23, wherein said determining comprises ordering said provided plurality points in time, and selecting a sufficient number of the least recently used allocations to be released to bring the aggregate allocations to at most a predetermined threshold.

26. (Original) The method of claim 23, wherein the resource monitor is a component of an application service provision apparatus.

27. (Previously Presented) An apparatus comprising:

storage medium having stored therein a plurality of programming instructions designed to implement a shared resource consumer, including the ability to:

accept allocations of a plurality of portions of a shared resource,

track points in time the allocations were last used,

receive a request to provide the tracked points in time,

provide, in response, the tracked points in time as requested,

receive instructions to release selected ones of the allocations, and

release the specified allocations as instructed; and

PAGE 9/13 * RCVD AT 3/22/2006 3:21:11 PM [Eastern Standard Time] * SVR:USPTO-EFXRF-5/22 * DNIS:2738300 * CSID:2062920460 * DURATION (mm-ss):04-02

at least one processor coupled to the storage medium to execute the programming instructions.

28. (Original) The apparatus of claim 27, wherein the apparatus is an application service provision apparatus, and the shared resource consumer is an application requiring application service provision runtime library support.

29. (Original) The apparatus of claim 27, wherein the apparatus is an application service provision apparatus, and the shared resource consumer is a function of an application service provision runtime library.

30. (Previously Presented) An apparatus comprising:

storage medium having stored therein a plurality of programming instructions designed to implement a shared resource monitor, including the abilities to:

conditionally request a plurality of shared resource consumers to provide corresponding tracked plurality points in time, where corresponding plurality of portions of a shared resource allocated to the plurality of shared resource consumers were last used, and

determine which if any of the plurality of allocations of the portions of the shared resource are to be released by the plurality of shared resource consumers, and instruct the plurality of shared resource consumers to release selected ones of the plurality of allocations accordingly; and

at least one processor coupled to the storage medium to execute the programming instructions.

31. (Previously Presented) The apparatus of claim 30, wherein said shared resource monitor is designed to make said conditional request when aggregate allocations of the shared resource reach a pre-determined threshold.

PAGE 10/13 * RCVD AT 3/22/2006 3:21:11 PM [Eastern Standard Time] * SVR:USPTO-EFXRF-5/22 * DNIS:2738300 * CSID:2062920460 * DURATION (mm-ss):04-02

32. (Previously Presented) The apparatus of claim 30, wherein said shared resource monitor is designed to perform said determining by ordering said provided plurality points in time, and selecting a sufficient number of the least recently used allocations to be released to bring an aggregate of the allocations to at most a predetermined threshold.

33. (Original) The apparatus of claim 30, wherein the apparatus is an application service provision apparatus, and the resource monitor is a component of said application service provision apparatus.

34.-37. (Cancelled)

PAGE 11/13 * RCVD AT 3/22/2006 3:21:11 PM [Eastern Standard Time] * SVR:USPTO-EFXRF-5/22 * DNIS:2738300 * CSID:2062920460 * DURATION (mm-ss):04-02